Amendments to the Specification:

Please replace paragraph [010] with the following amended paragraph:

One approach to forming packets for a memory hub system that has been proposed will now be explained with reference to Figure 2 in which several 32-bit groups of data from each of several memory accesses or "transactions" are shown in the right hand side of Figure 2. Transaction T0 consists of 7 32-bit groups D0-D6 of data, which are coupled to a data organization unit 160 on a 96-bit bus 162. The bus 162 is therefore capable of coupling three 32-bit groups of data to the data organization ~~162~~160 each cycle of a core clock CCLK signal, *i.e.*, a clock signal that is used internally in the memory hubs 140. Transaction T1 also consists of 7 32-bit groups D0-D6 of data, and it is coupled to a data organization unit 160 on a 64-bit bus 164, which is capable of coupling two 32-bit groups of data to the data organization ~~162~~160 each CCLK cycle. Transaction T2 consists of only 5 32-bit groups D0-D4 of data, and it is also coupled to a data organization unit 160 on a 64-bit bus 166 two 32-bit groups each CCLK cycle. Finally, transaction T3 consists of 12 32-bit groups D0-D11 of data, and it is coupled to a data organization unit 160 on a 128-bit bus 168, which is capable of coupling four 32-bit groups of data to the data organization ~~162~~160 each CCLK cycle. It can therefore be seen that components in the memory hub 140 outputting data on respective buses having different widths can interface with the data organization unit 160.

Please replace paragraph [024] with the following amended paragraph:

A method of forming packets for a memory hub system according to one example of the present invention will now be explained with reference to Figure 3. As shown in Figure 3, several 32-bit groups of data from each of several memory accesses or "transactions" are identical to those shown in Figure 2 for purposes of illustrating the differences therebetween except that a portion of an additional transaction T4 is shown in Figure 3. Transaction T4 consists of only 4 32-bit groups D0-D3 of data, where two 32-bit groups D0-D3 of data are clocked on a 64-bit bus 169. As before, transaction T0 consists of 7 32-bit groups of data D0-D6, transaction T1 also consists of 7 32-bit groups of data D0-D6, transaction T2 consists of 5 32-bit groups of data D0-D4, and transaction T3 consists of 12 32-bit groups of data D0-D11. Additionally, transaction T4 consists of 4 32-bit groups of data D0-D3.

Please replace paragraphs [030] and [031] with the following amended paragraphs:

The interfaces 210-216 are coupled to a switch 260 through a plurality of bus and signal lines, represented by buses 228. The buses 228 are conventional, and include a write data bus coupled to the receiver interfaces 210, ~~224~~214 and a read data bus coupled to the transmit interfaces 212, ~~222~~216.

The switch 260 is coupled to four memory interfaces 270a-d which are, in turn, coupled to the memory devices ~~160~~148 (Figure 1). By providing a separate and independent memory interface 270a-d for each set of memory devices ~~160~~148, the memory hub 200 avoids bus or memory bank conflicts that typically occur with single channel memory architectures. The switch 260 is coupled to each memory interface through a plurality of bus and signal lines, represented by buses 274. In addition to coupling the interfaces 210-216 to the memory interfaces, the switch 260 can also couple the memory interfaces 210-216 to each other to allow memory packets to be coupled downstream or upstream through the memory module 130 to either another memory module 130 or the memory hub controller 128.

Please replace paragraph [033] with the following amended paragraph:

The switch 260 can be any of a variety of conventional or hereinafter developed switches. For example, the switch 260 may be a cross-bar switch or a set of multiplexers that do not provide the same level of connectivity as a cross-bar switch but nevertheless can couple the bus interfaces 210-216 to each of the memory interfaces ~~470~~270a-d. The switch 260 may also include arbitration logic (not shown) to determine which memory accesses should receive priority over other memory accesses. Bus arbitration performing this function is well known to one skilled in the art.

Please replace paragraphs [039], [040] and [041] with the following amended paragraphs:

The memory modules 130 are shown coupled to the memory hub controller 128 in a point-to-point coupling arrangement in which each portion of the high-speed buses ~~132,~~134 are coupled only between two points. However, it will be understood that other topologies may also be used. For example, it may be possible to use a multi-drop arrangement in which a single downstream bus (not shown) and a single upstream bus (not shown) are coupled to all of the memory modules 130. A switching topology may also be used in which the memory hub controller 128 is selectively coupled to each of the memory modules 130 through a switch (not shown). Other topologies that may be used will be apparent to one skilled in the art.

One embodiment of the data organization system 220 used in the memory hub 200 of Figure 4 is shown in Figure 5. The data organization system 220 can also be used in the memory hub controller 128 to couple data to the high-speed downstream bus 222. The portions of receive interfaces 210, 224 (Figure 4) and a receive interface in the memory hub controller 128 that capture the memory packets from the high-speed buses ~~132,~~134 is relatively straightforward, and the design of a suitable system is well within the ability of one skilled in the art.

The data organization system 220 includes a data buffer 230 that receives the 32-bit groups of data that are to be coupled through the high-speed buses ~~132,~~134. In the case of the data organization system 220 in the memory hub controller 128, the source of the data may be the processor 104 (Figure 1) or any other memory access device. In the case of the memory modules 130, the data may originate from the memory devices 148 in the memory modules 130 or from another memory module 130. In any case, the groups of data are clocked into the data buffer 230 responsive to the core clock signal, as indicated schematically in Figure 5. As also schematically shown in Figure 5, the data stored in the data buffer 230 for different transactions are of different lengths.